

Patientenverwaltung

**von Jasmin Sölling, Darius Keil, Jennifer Ried,
Julian Bermel und Timo Seuser**

Inhaltsverzeichnis

1. Dokumentation

- a. Patient als Objekt
- b. Patient hinzufügen
- c. Patientendaten bearbeiten
- d. Patientensuche
- e. Patient anzeigen
- f. Patient löschen
- g. Patientenliste
- h. interne Patienten Speicherung
- i. Export ausgewählter Patienten
- j. Fehlermeldungen

2. Bedienungsanleitung

- a. Patient hinzufügen
- b. Patientendaten bearbeiten
- c. Patientensuche
- d. Patient löschen
- e. Patient anzeigen
- f. Daten exportieren
- g. Patientenliste

3. Datenflussdiagramm

4. 4+1 Sichtenmodell

- a. Klassendiagramm
- b. Aktivitätsdiagramm
- c. Komponentendiagramm
- d. Verteilungsdiagramm
- e. Anwendungsfalldiagramm

5. Quellen

1.Dokumentation

Patient als Objekt

Die Patienten werden in Form von Patient Objekten verwaltet. Die Attribute eines Patienten Objekts sind Titel, Nachname, Vorname, Geburtsdatum, Geschlecht, Straße, Hausnummer, Postleitzahl, Ort, Telefonnummer und ID. Dem Objekt wird vom System eine eindeutige ID gegeben, um zwischen den Objekten eindeutig unterscheiden zu können, dies ist besonders beim Löschen und dem Ändern der Daten wichtig.

Patienten hinzufügen

Wird im MainWindow das Plus Icon gedrückt, öffnet sich ein neues Fenster zum hinzufügen der Patienten. Die ersten Felder befassen sich mit dem Titel, dem Nachnamen und dem Vornamen. Danach wird nach dem Geburtsdatum gefragt. Hier erscheint sobald der Benutzer auf das Feld klickt ein Kalender, indem der Benutzer ganz bequem das richtige Datum anklicken kann. Als Standard wird das Datum 01.01.2000 angezeigt. Anschließend kann der Nutzer das Geschlecht aus einer ComboBox auswählen. Die Auswahlmöglichkeiten sind "w" für weiblich, "m" für männlich und "d" für divers. Unterhalb der ComboBox befindet sich das Feld um die Telefonnummer einzutragen. Die Telefonnummer muss mit einer "0" begonnen werden und es werden nur Zahlen akzeptiert. Darunter befinden sich die Felder, in denen der Nutzer, die Adresse in Form von Straße, Hausnummer, Postleitzahl und Ort eingibt. Die Postleitzahl muss 5 Ziffern lang sein und es werden auch hier, wie bei der Telefonnummer nur Ziffern akzeptiert. Die eingegebenen Daten werden dynamisch in einem Block rechts neben den Feldern angezeigt, was das überprüfen der Daten vereinfachen soll. Das Geschlecht wird nicht wie in der Auswahl zusehen als einzelner Buchstabe angezeigt, sondern als Wort ausgeschrieben. Der Nutzer hat die Wahl den Vorgang abubrechen oder die Daten zu speichern .

Die Daten werden aus den TextEdit Feldern ausgelesen und in QString Variablen temporär gespeichert. Ausgenommen davon sind die Hausnummer und die Postleitzahl, diese werden in int Variablen temporär gespeichert. Das Geburtsdatum wird als QDate Variable temporär gespeichert. Das Geschlecht wird aus der ComboBox ausgelesen und im Format eines QStrings gespeichert. Anschließend wird überprüft welche Auswahl getroffen wurde und das zu dem Buchstaben passende Geschlecht in ausgeschriebener Form temporär in einem QString gespeichert wird. Die temporären Variablen bilden nun alle die Attribute des Objekt Patient.

Patientendaten bearbeiten

Der Nutzer hat die Möglichkeit die Daten eines Patienten zu ändern. Hierzu wählt er die Zeile des Patienten aus und klickt auf das Stift-Icon. Danach öffnet sich das gleiche Fenster, wie es beim Patienten hinzufügen zu sehen ist. Allerdings sind die Felder bereits ausgefüllt mit den bisherigen Daten. Erneut hat der Benutzer die Möglichkeit den Vorgang abubrechen oder die Daten zu speichern.

Patientensuche

Beim Klicken auf das Suchen-Icon wird eine Suchleiste sichtbar. Diese beinhaltet ein Feld für den Vornamen, eines für den Nachnamen, eines für das Geburtsdatum und zwei Buttons um die Suche auszuführen oder diese zu beenden. Der Benutzer kann diese Felder beliebig ausfüllen, es muss jedoch mindestens ein Feld ausgefüllt sein. Das Geburtsdatum muss im Format "yyyy.mm.dd" eingegeben werden. Die Felder können bei der Suche beliebig kombiniert werden. Im Anschluss daran drückt man auf den "Suchen"-Button und der gesuchte Patient wird angezeigt. Drückt man daraufhin den "Suchen beenden"-Button werden die anderen Patienten wieder angezeigt und die Suchleiste wird wieder versteckt.

Patient anzeigen

Der Benutzer kann sich die genauen Daten des Patienten auf dem MainWindow anzeigen lassen. Hierfür klickt er einen Patienten aus der Liste an, sodass dieser blau markiert ist und klickt dann auf den Button mit der Person. Nun erscheinen rechts von der Liste die genauen Daten des ausgewählten Patienten. Diese Anzeige kann durch das klicken des X-Symbols, dass sich rechts unterhalb des Export-Buttons befindet, wieder beendet werden. Durch die Anzeige verkleinert sich die Breite der Liste. Wird die Anzeige beendet erscheint die Liste wieder in derselben Größe wie beim Start des Programms.

Patient löschen

Patienten können nur einzeln gelöscht werden. Dafür die Zeile des Patienten auswählen und auf das Mülleimer-Icon klicken. Anschließend erscheint eine MessageBox, in der der Benutzer gefragt wird, ob er den Patienten tatsächlich löschen möchte. Erst wenn er diese Frage per Button bestätigt, wird der Patient aus der internen Speicherung gelöscht. Der Nutzer hat aber auch die Möglichkeit den Löschvorgang abubrechen, indem er die Frage verneint.

Patientenliste

Die Tabelle wird in einem dynamischen QTableWidgetItem dargestellt, das sich jeweils der Fenstergröße anpasst dabei aber auch eine minimale Fenstergröße besitzt, um zu gewährleisten, dass alle Elemente sichtbar sind. Die Daten für das QTableWidgetItem werden durch QList Patient bereitgestellt. Jede einzelne Zelle wird durch ein jeweiliges eigenes QTableWidgetItemItem gefüllt. Die Zeilen werden von einem Patienten gefüllt. Es gibt so viele Zeilen wie Patientendaten vorhanden sind. Das alphabetische Sortieren der Tabelle wird durch die Eigenschaft "sortingEnabled" des QTableWidgetItem gewährleistet.

Interne Patienten Speicherung

Die Patienten werden intern in einer QMap gespeichert, wobei der Schlüssel die Patienten ID und der Wert ein Patient Objekt ist. Bei der Klasse des internen Speichers wurde das Singleton-Pattern angewandt. Dieses sorgt dafür, dass nur eine Instanz der Klasse existiert und somit auch nur eine Instanz der QMap, also der internen Speicherung. Diese Klasse verfügt weiterhin über Funktionen zum Löschen und Aktualisieren / Neuerstellung eines Patienten auf Basis der Patienten ID. Auch gibt es eine Funktion, die alle Patienten als Liste übergibt, jedoch kann man die Patienten nur einzeln ändern.

Zusätzlich wurde das Observer-Pattern genutzt, um der Tabelle, die die Patienten anzeigt, mitzuteilen, dass es eine Aktualisierung, sowohl bei Änderung als auch bei Löschung eines Patienten, gab.

Die Patientenliste wird nach jeder Aktualisierung oder Löschung neu in der "Patienten.json" - Datei gespeichert. Diese ist auch die Grundlage auf der die Tabelle bei einem Neustart gefüllt wird. Dabei muss die Datei in demselben Ordner liegen in dem das Projekt ausgeführt wird und wird auch ebenfalls dort gespeichert. Das Patient Objekt hat dabei eine eigene Funktion für jeweils das Erstellen eines json-Objekt aus den Eigenschaften des Patienten und das Umgekehrte herausholen der Eigenschaften aus dem json-Objekt.

Export ausgewählter Patienten

In der Tabelle können einzelne oder auch mehrere Patienten ausgewählt werden, sodass ihre Daten in einer gesonderten "Export.json" - Datei gespeichert werden. Wie schon bei der internen Speicherung beschrieben, besitzt das Patient Objekt über eine Funktion um die Eigenschaften eines Patienten in ein json-Objekt umzuwandeln. Die QTableView kann eine QList in der alle ausgewählten Zeilen

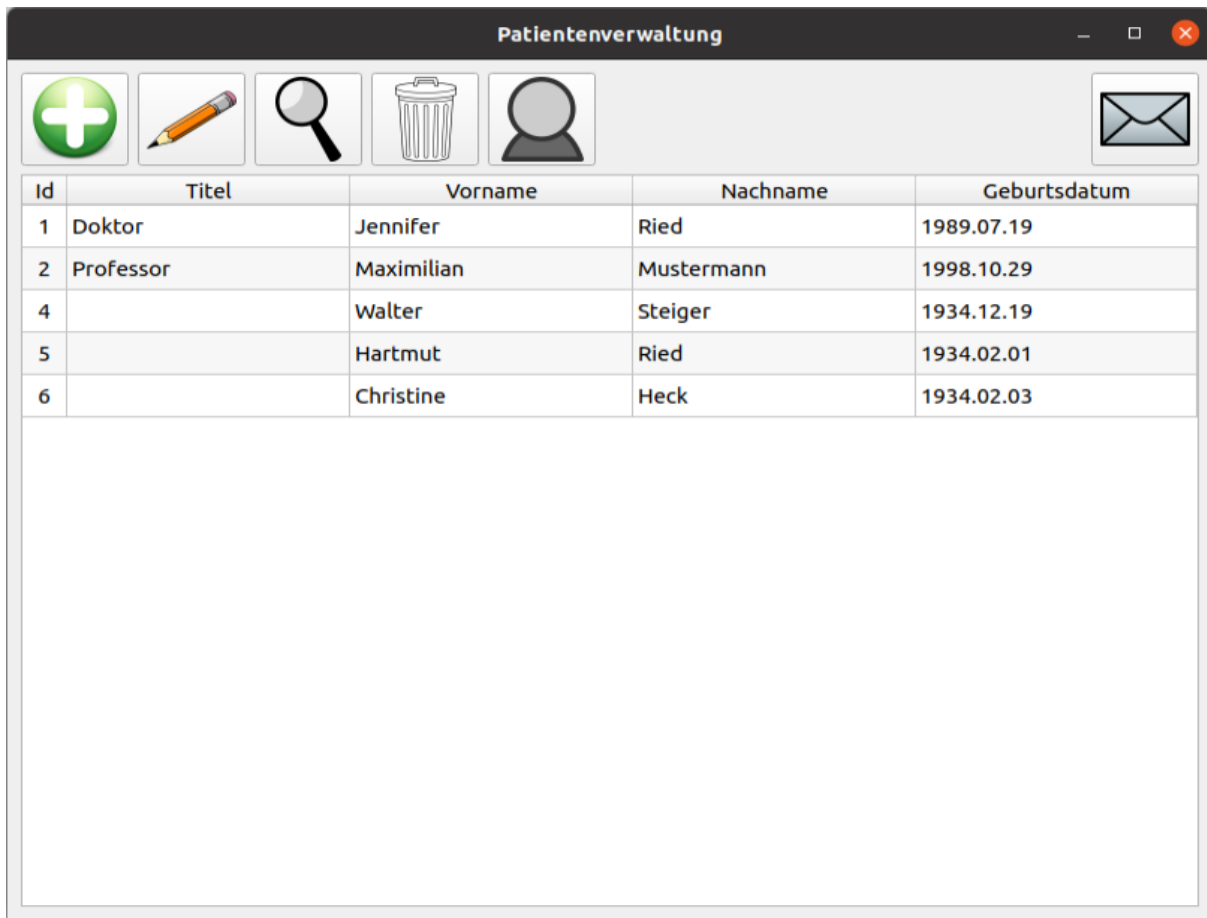
aufgelistet sind ausgegeben. Diese Liste wird in einer for-Schleife abgearbeitet und für jeden Patienten ein eigenes json-Objekt angelegt und in einem jsonDokument-Objekt gespeichert. Existiert die "Export.json"-Datei bereits, gibt es eine Abfrage, ob man diese Datei überschreiben möchte, wenn man dem zustimmt, wird die Datei überschrieben und die zuvor ausgewählten Zeilen sind nun nicht mehr vorausgewählt. Zudem gibt es eine Erfolgsmeldung, wenn die json-Datei erfolgreich gespeichert wurde. Wird die Abfrage verneint, bleibt die Auswahl erhalten, aber die Patienten werden nicht in der Datei gespeichert. Man muss selbstständig den Namen der Datei ändern, wenn man die vorige "Export.json"-Datei behalten und zusätzlich die ausgewählten Patienten exportieren möchte. Auch diese Datei wird in demselben Ordner gespeichert in dem das Projekt ausgeführt wird.

Fehlermeldungen

Um falsche Eingaben und Probleme festzustellen, wurden mehrere Fehlermeldungen implementiert. In den folgenden Szenarien wird eine Fehlermeldung über eine QMessageBox ausgegeben:

- ungültige Eingaben(z.B Zahlen in einem Namen) bei Erstellung oder Änderung eines Patienten
- unzulässige Länge der Postleitzahl bei Erstellung oder Änderung eines Patienten
- fehlende Eingaben bei Erstellung oder Änderung eines Patienten
- mehr als ein Patient ausgewählt beim löschen, ändern und anzeigen
- keinen Patienten ausgewählt beim löschen, ändern, anzeigen und export
- bei der Suche wurde kein Suchkriterium eingegeben
- eine json-Datei konnte nicht geöffnet werden bei Export, Start des Programmes oder Zwischenspeicherung bei Änderung eines Patienten

2. Bedienungsanleitung



Nach dem Start des Programms öffnet sich die Benutzeroberfläche.

Oben links befinden sich fünf Buttons:



Patient hinzufügen



Daten eines ausgewählten Patienten bearbeiten



Suchfunktion



Ausgewählten Patienten löschen



Patientendaten vollständig anzeigen

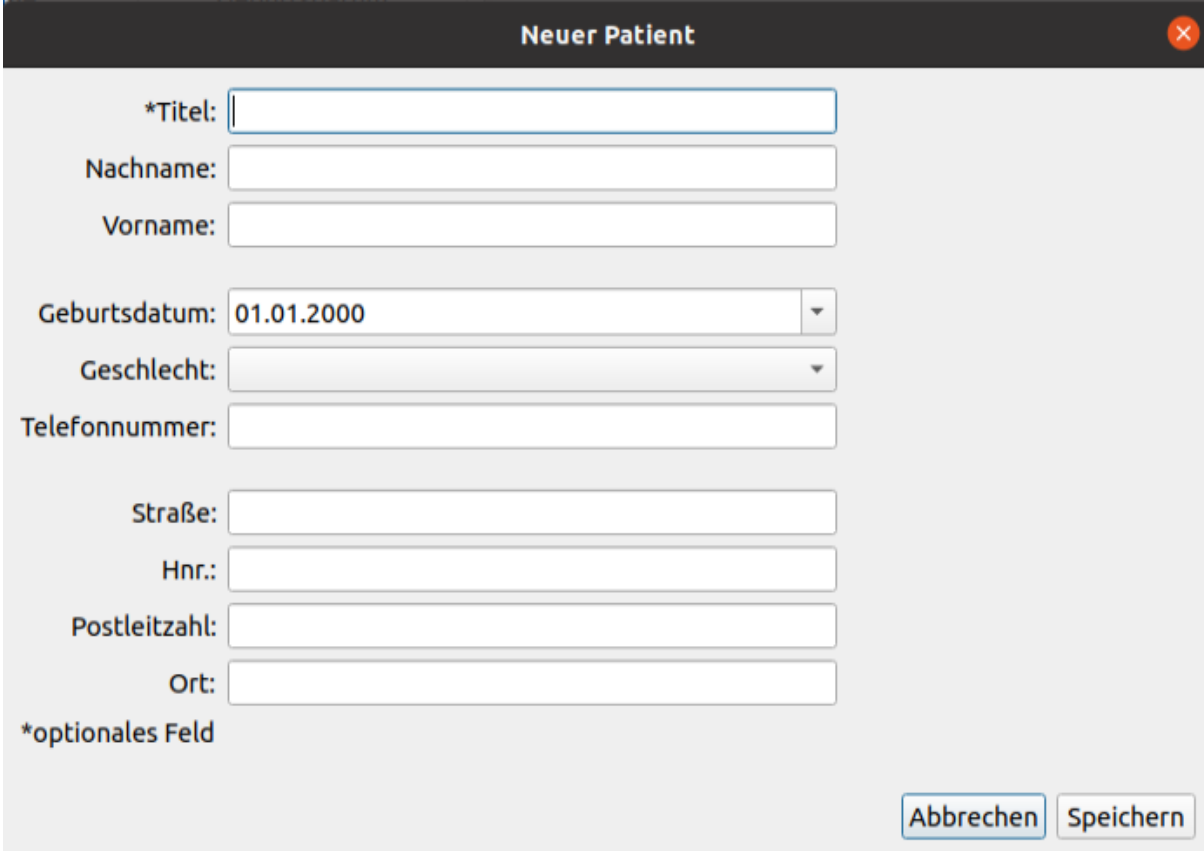
Oben rechts befindet sich ein Button:



Patientenliste exportieren

Patient hinzufügen:

Klicken Sie den  Button. Danach öffnet sich ein weiteres Fenster.




The screenshot shows a web form titled "Neuer Patient" with a dark header bar containing the title and a close button (X). The form fields are as follows:

- *Titel:
- Nachname:
- Vorname:
- Geburtsdatum: (with a dropdown arrow)
- Geschlecht: (with a dropdown arrow)
- Telefonnummer:
- Straße:
- Hnr.:
- Postleitzahl:
- Ort:

Below the fields is the text "*optionales Feld". At the bottom right are two buttons: "Abbrechen" and "Speichern".


Tragen Sie die entsprechenden Daten in die dafür vorgesehen Felder ein. Das Feld "Titel" ist optional. Für die Datumseingabe öffnen Sie rechts das Dropdown-Menü und wählen Sie das entsprechende Datum aus dem Kalender PopUp das erscheint aus, oder geben Sie die Daten per Tastatur im Format "DD.MM.YYYY" ein. Für die Auswahl des Geschlechts öffnen Sie ebenfalls ein Dropdown-Menü und wählen Sie das entsprechende Geschlecht aus. Es stehen 3 Geschlechter zur Auswahl: "w" für weiblich, "m" für männlich und "d" für divers. Rechts neben den Eingabefeldern wird immer eine aktuelle Gesamtübersicht der Daten angezeigt. Hinter dem Datum steht in Klammern das Alter des Patienten, dies ist abhängig von dem eingetragenen Datum. Nachdem Sie alle gewünschten Daten eingetragen haben, drücken Sie den „Speichern“-Button unten rechts. Falls Sie das Fenster verlassen wollen, ohne dass die Daten gespeichert werden, drücken Sie „Abbrechen“.

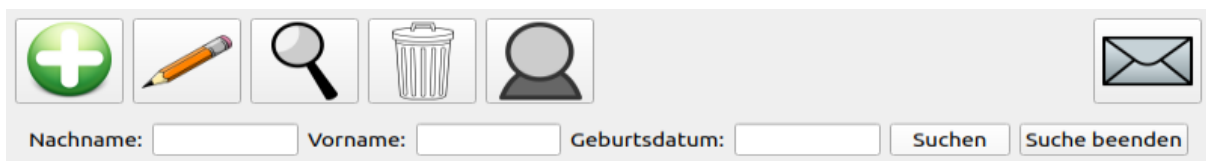
Patientendaten bearbeiten:

Wählen Sie einen bereits vorhandenen Patienten aus der Liste durch einmaliges Klicken aus. Die entsprechende Zeile wird blau markiert. Klicken Sie den  Button. Danach öffnet sich ein weiteres Fenster.

Alle vorhandenen Daten des Patienten werden angezeigt und können frei bearbeitet werden. Datum- und Geschlechtsauswahl funktionieren genau wie bei „Patient hinzufügen“. Nachdem Sie alle gewünschten Daten bearbeitet haben, drücken Sie den „Speichern“-Button unten rechts. Falls Sie das Fenster verlassen wollen, ohne dass die Daten gespeichert werden, drücken Sie „Abbrechen“.

Patientensuche:

Klicken Sie den  Button. Daraufhin wird eine Suchleiste unter der Icon-Leiste sichtbar.




The image shows a search interface for patients. It features a horizontal toolbar with five icons: a green circle with a white plus sign, a pencil, a magnifying glass, a trash can, and a person silhouette. To the right of these icons is an envelope icon. Below the icons are three input fields labeled 'Nachname:', 'Vorname:', and 'Geburtsdatum:'. To the right of these fields are two buttons: 'Suchen' and 'Suche beenden'.

Tragen Sie den Namen oder das Geburtsdatum des Patienten in die dafür vorhergesehenen Felder ein. Bitte beachten Sie, dass mindestens eines der Felder ausgefüllt sein muss. Das Geburtsdatum muss im Format "YYYY.MM.DD" eingegeben werden. Im Anschluss betätigen Sie den "Suchen"-Button und die Treffer passend zu Ihrer Suche werden Ihnen angezeigt. Wenn Sie die anderen Patienten wieder einblenden möchten, betätigen Sie den "Suche beenden"-Button. Das Betätigen des "Suche beenden"-Buttons blendet ebenfalls die Suchleiste wieder aus.

Patient löschen:

Wählen Sie einen bereits vorhandenen Patienten aus der Liste durch einmaliges

Klicken aus. Die entsprechende Zeile wird blau markiert. Klicken Sie den  Button. Es öffnet sich ein Fenster, in dem Sie den Löschvorgang durch Klicken von „Yes“ bestätigen, oder durch „No“ abbrechen können.

Patient anzeigen:

Wählen Sie einen bereits vorhandenen Patienten aus der Liste durch einmaliges


Klicken aus. Klicken Sie den  Button.

Rechts neben der Tabelle werden nun die vollständigen Daten des Patienten angezeigt.

Durch Klicken des  Buttons können sie diese Anzeige wieder entfernen.

Daten exportieren:

Wählen Sie alle Patienten aus, die Sie exportieren wollen.

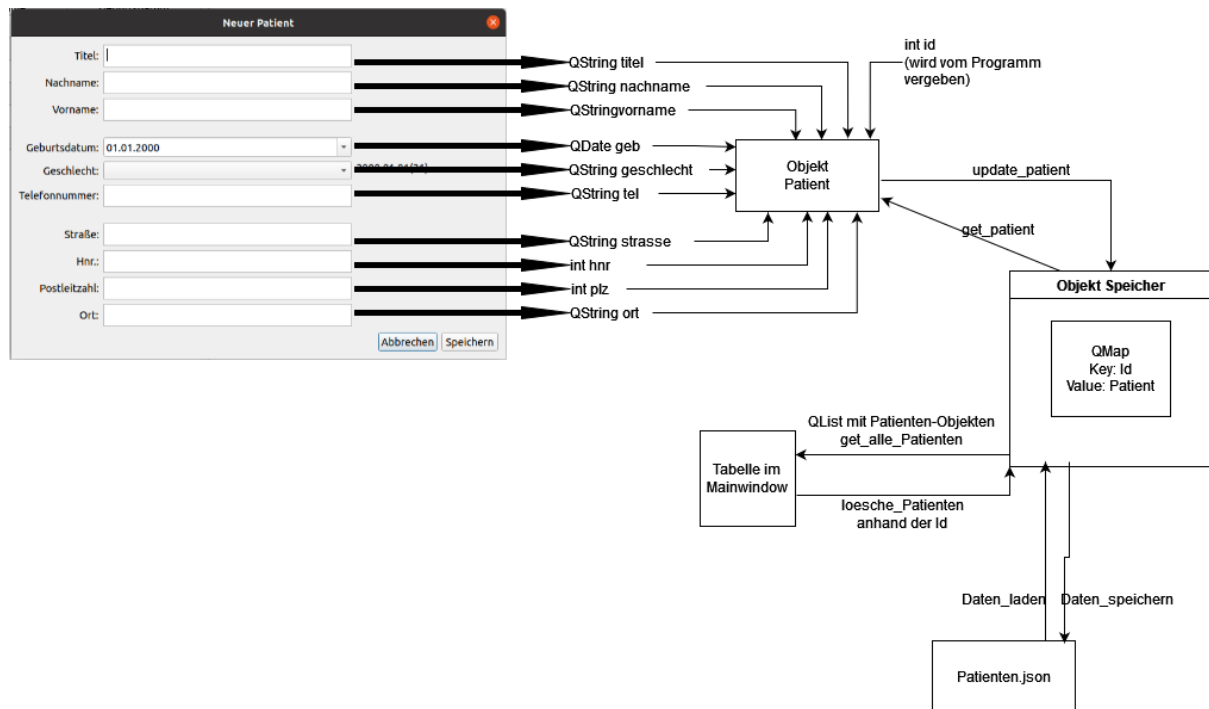
Klicken sie dann den  Button. Wurde bereits einen Export durchgeführt und die Datei nicht umbenannt kommt eine Hinweismeldung, ob der alte Export überschrieben werden soll. Wird auf „Yes“ gedrückt, so wird der alte Export überschrieben und der neue in die Datei „Export.json“ geschrieben. Wenn „no“ ausgewählt wird kommt man ins Hauptfenster zurück und es hat kein Export stattgefunden.

Patientenliste:

Hier finden Sie immer alle im System gespeicherten Patienten. Jeder Patient hat seine eigene ID, wodurch jeder Patient klar zuzuordnen ist. Sie ist die erste Spalte der Tabelle. Zusätzlich wird immer „Titel“, „Vorname“, „Nachname“ und „Geburtsdatum“ angezeigt. Sie können durch klicken der jeweiligen Spalten-Überschriften die Zeilen alphabetisch von A – Z oder Z – A sortieren. Falls mehr Patienten eingetragen sind, als im Fenster Platz vorhanden ist, können Sie die Liste scrollen oder durch Verschiebung des Scroll bar Balkens bewegen.

3.Datenflussdiagramm

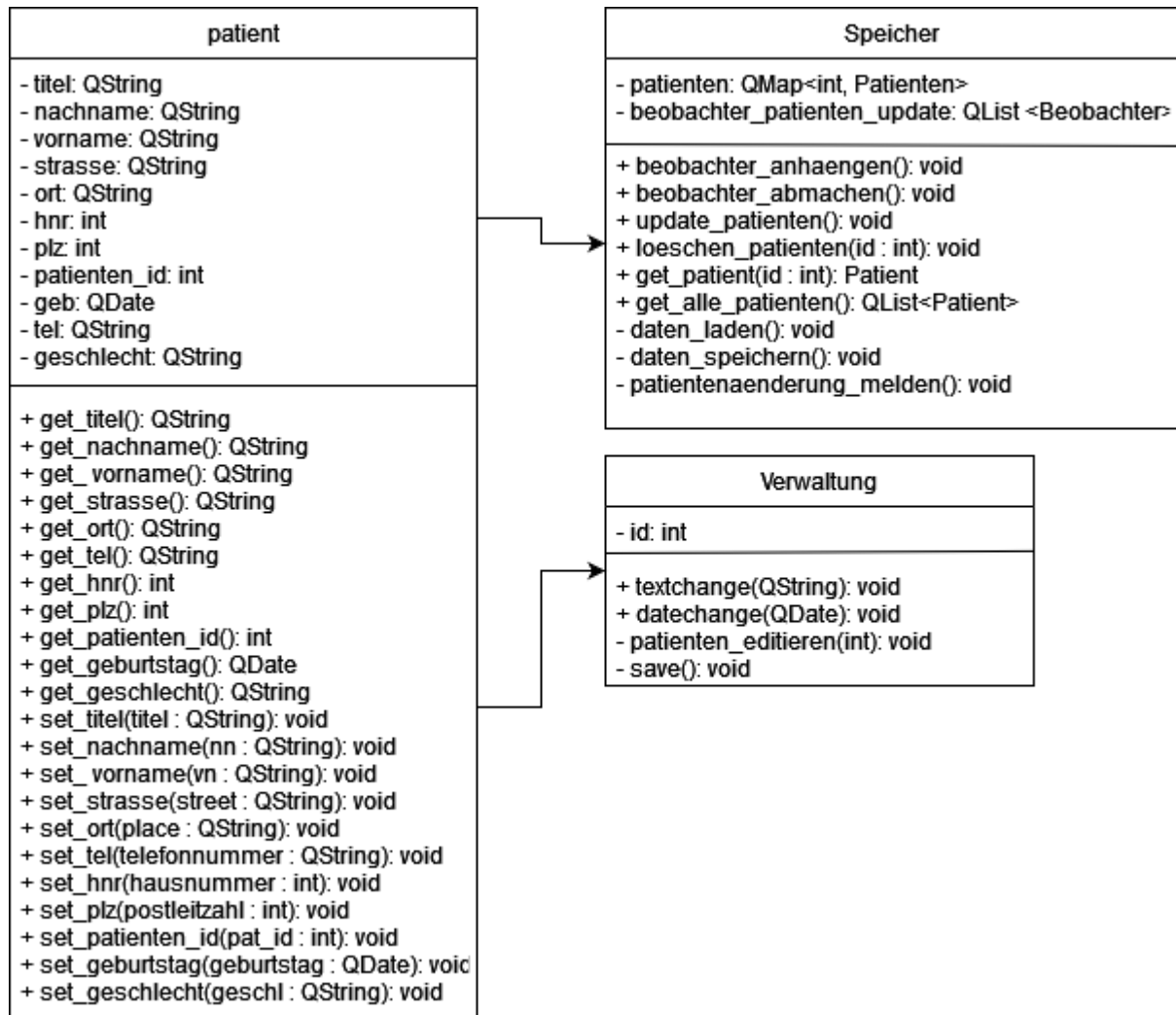
Datenflussdiagramm



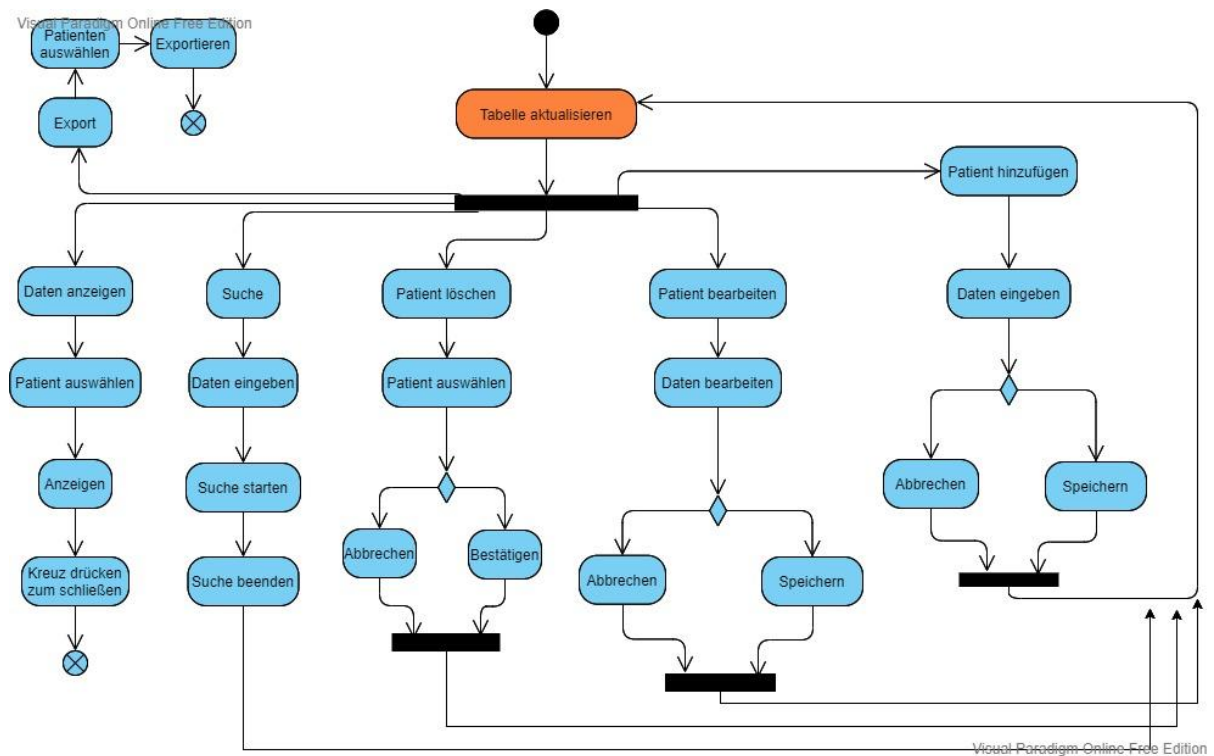
Der Benutzer füllt die Felder des “Neuer Patient” Fensters aus, wobei der Titel ein optionales Feld ist. Die eingegebenen Daten werden in QString, QDate und integer Variablen gespeichert. Die Werte der Variablen erzeugen zusammen mit der vom Programm vergebenen ID, die ebenfalls eine integer Variable ist, ein neues Objekt der Klasse Patient. Diese wird durch die Funktion “update_Patient” der internen Speicherung hinzugefügt. Die interne Speicherung wird durch eine QMap realisiert. Anhand der Patienten ID wird ein Patienten zur Bearbeitung dem Benutzer wieder übergeben. Zudem werden die Daten der Patienten in einer json-Datei gespeichert und aus ihr wird zu Beginn des Programms die Patienten ausgelesen. Auch wird die Tabelle des MainWindows mit den Werten der QMap gefüllt. Wird ein Patient gelöscht wird dieser anhand der Patienten ID aus der internen Speicherung entfernt.

4. 4+1 Sichtenmodell

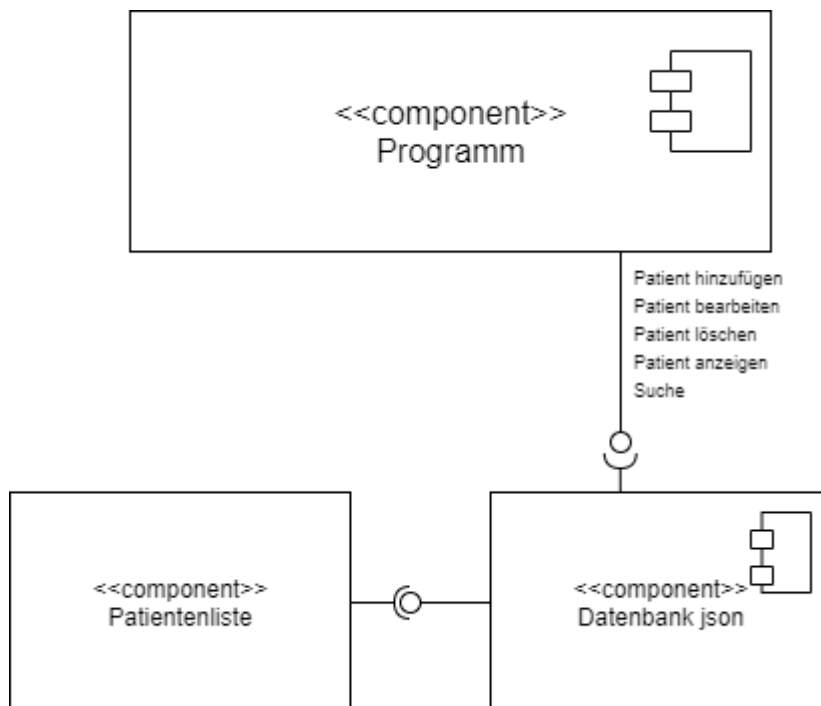
Klassendiagramm (logische Sicht)



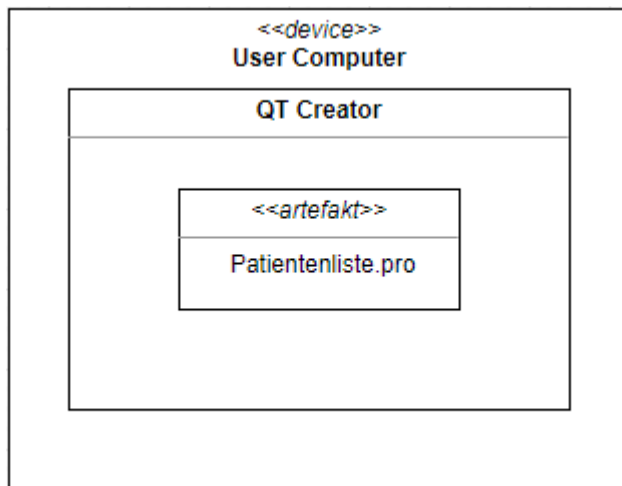
Aktivitätsdiagramm (Prozess Sicht)



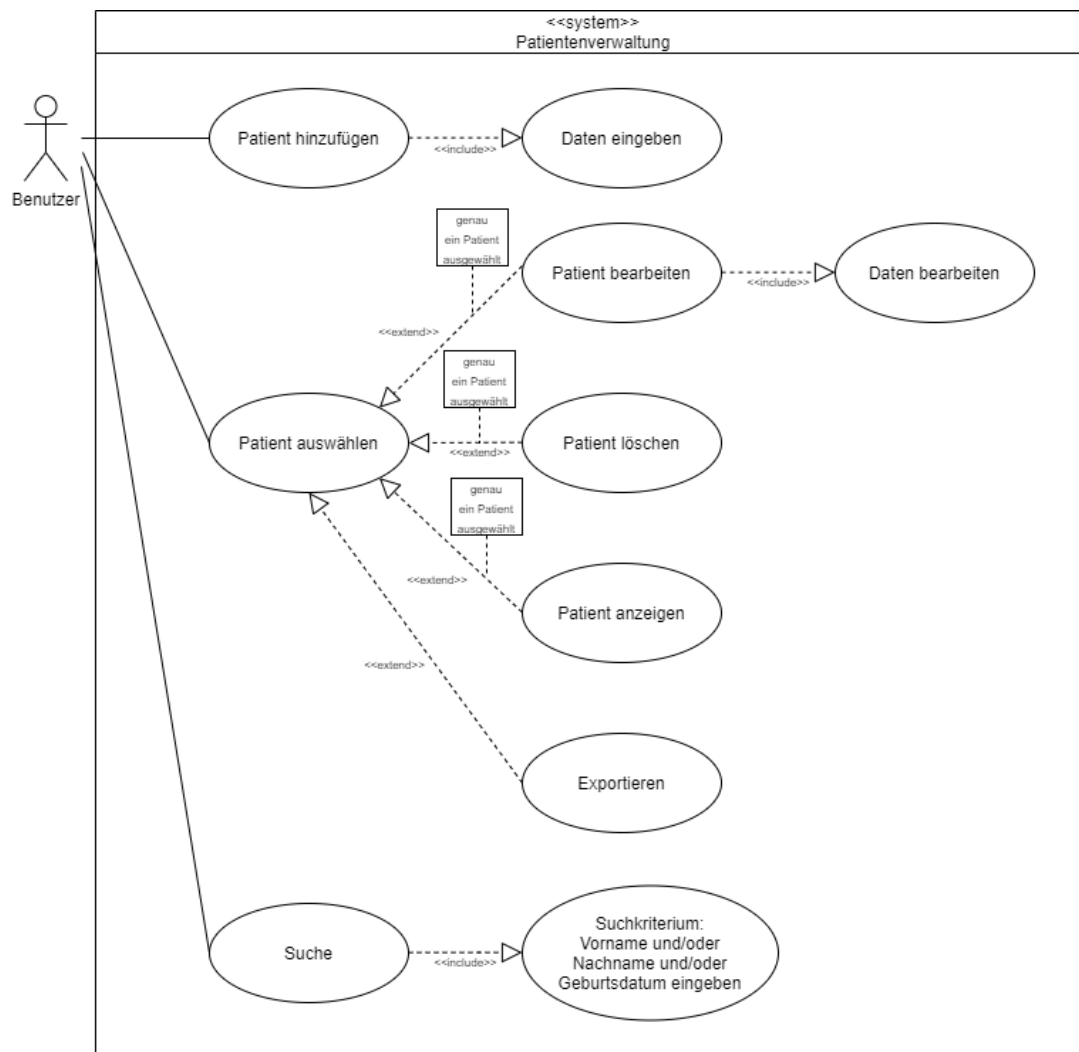
Komponentendiagramm (Entwicklungs Sicht)



Verteilungsdiagramm (physische Sicht)



Anwendungsfalldiagramm (Szenario Sicht)



5. Quellen

- qtforum
- stackoverflow
- <https://doc.qt.io> (offizielle Dokumentation zu Qt)
- <https://www.ionos.de/digitalguide/websites/web-entwicklung/was-ist-das-singl-eton-pattern/> (Singleton-Pattern)
- <https://riptutorial.com/cplusplus/example/24695/observer-pattern>
(Observer-Pattern)